

1 **CLAIMS**

2 What is claimed is:

3 1. A computer-implemented method for testing a

4 parameterizable logic core, comprising:

5 randomly generating a set of parameter values for the
6 logic core;7 generating a netlist from the set of parameter values
8 and logic core; and

9 simulating circuit behavior with the netlist.

10

11 2. The method of claim 1, further comprising for each
12 parameter:13 generating a random parameter value within
14 predetermined upper and lower limits associated with the
15 parameter; and16 generating a new random parameter value if the random
17 parameter value fails to meet predetermined criteria.

18

19 3. The method of claim 2, further comprising:

20 assigning respective probabilities to one or more
21 numbers between the upper and lower limits for one or more
22 of the parameters; and23 generating the random parameter value as a function of
24 the probabilities.

25

26 4. The method of claim 3, further comprising:

27 providing the parameter value as input to a graphical
28 user interface;29 generating random replacement values for invalid
30 parameter values detected by the graphical user interface.

31

32 5. The method of claim 1, further comprising:

33 providing the set of parameter values to a graphical
34 user interface; and35 identifying invalid parameter values with the graphical
36 user interface.

1

2 6. The method of claim 5, further comprising:
3 generating random replacement parameter values for the
4 invalid parameters; and
5 repeating the steps of providing the replacement values
6 to the graphical user interface, identifying invalid
7 parameter values, and generating random replacements until
8 all the parameter values are valid.

9

10 7. The method of claim 5, further comprising:
11 selecting a random order in which to provide parameter
12 values to the graphical user interface; and
13 providing the parameters one-by-one as input to the
14 graphical user interface.

15

16 8. The method of claim 5, further comprising:
17 cloning the set of parameter values;
18 mutating the set of parameter values, whereby a mutated
19 set of parameter values is produced;
20 generating a new netlist from the mutated set of
21 parameter values and logic core; and
22 simulating circuit behavior with the new netlist.

23

24 9. The method of claim 8, wherein the set of parameter
25 values is cloned and mutated only if an error is detected in
26 simulating the circuit behavior.

27

28 10. The method of claim 1, further comprising:
29 cloning the set of parameter values;
30 mutating the set of parameter values, whereby a mutated
31 set of parameter values is produced;
32 generating a new netlist from the mutated set of
33 parameter values and logic core; and
34 simulating circuit behavior with the new netlist.

35

1 11. The method of claim 10, wherein the set of parameter
2 values is cloned and mutated only if an error is detected in
3 simulating the circuit behavior.

4
5 12. The method of claim 11, further comprising repetitively
6 cloning and mutating sets of parameters when errors are
7 detected in simulating the circuit behavior, whereby
8 multiple generations of sets of parameters are created.

9
10 13. The method of claim 12, wherein one or more of the
11 parameter values in a parameter set are mutated.

12
13 14. The method of claim 13, wherein a number of parameter
14 values mutated in a set of parameters is a function of a
15 number of generations previously created.

16
17 15. The method of claim 1, further comprising:
18 identifying which parameters of a graphical user
19 interface are editable; and
20 providing the set of parameter values to the graphical
21 user interface.

22
23 16. The method of claim 15, further comprising:
24 selecting a random order in which to provide the
25 parameter values to the graphical user interface; and
26 providing the values one-by-one as input to the
27 graphical user interface.

28
29 17. The method of claim 1, further comprising:
30 accumulating respective numbers of tests having been
31 performed using different parameter values;
32 accumulating respective numbers of tests failed using
33 each of the parameter values; and
34 highlighting parameters having numbers of failed tests
35 equal to the number of tests.

36

1 18. A system for testing a parameterizable logic core,
2 comprising:
3 a test controller configured and arranged to randomly
4 generate a set of parameter values for the logic core;
5 a core generator coupled to the test controller, the
6 core generator configured and arranged to generate a netlist
7 from the logic core and set of parameter values; and
8 a simulator coupled to the test controller, the
9 simulator configured and arranged to simulating circuit
10 behavior with the netlist and a predetermined test bench.

11
12 19. A system for testing a parameterizable logic core,
13 comprising:
14 a test controller including a GUI-driver, the GUI-
15 driver configured and arranged to randomly generate a set of
16 parameter values for the logic core;
17 a core generator including a GUI coupled to the GUI-
18 driver, the core generator configured and arranged to
19 generate a netlist from the logic core and set of parameter
20 values; and
21 a simulator coupled to the test controller, the
22 simulator configured and arranged to simulate circuit
23 behavior with the netlist and a predetermined test bench.

24
25 20. An apparatus for testing a parameterizable logic core,
26 comprising:
27 means for randomly generating a set of parameter values
28 for the logic core;
29 means for generating a netlist from the set of
30 parameter values and logic core; and
31 means for simulating circuit behavior with the netlist.

32
33
34